A Sparse Boosting Algorithm for Regression Problem

Po-Hsuan (Cameron) Chen **Yingfei Wang**

Abstract

We proposed a boosting regression algorithm that seeks to minimize the loss function while only using a sparse amount of weak predictor. We are interested in the case that the number of weak predictor, such as linear functions, step functions, quadratic functions, etc, is much larger than the number of training data we have. In this case, an unconstraint loss minimization is inadequate because under the scenario of aggregating the weak predictor, the composite predictor is likely to overfit the data. We imposed sparsity constraint on the output of gradient boosting [9] to select a sparse linear model for regression problem. The algorithm can be viewed as a coordinate descent method for the $l1$-regularized of the loss function used in the gradient boosting algorithm.

1 Introduction

Boosting is a generic modeling approach which has attracted a lot of attention in the machine learning, data mining and statistics communities. Boosting can most generally be described as a method for iteratively building an additive model $F(x) = \sum_j \alpha_j h_j(x)$. The essence of boosting is to design a way to adaptively select the next increment at each step to improve the fit. If the number of weak predictor, such as linear functions, step functions, quadratic functions, etc, is much larger than the number of training data we have, an unconstraint loss minimization is inadequate because under the scenario of aggregating the weak predictor, the composite predictor is likely to overfit the data. Hence, various regularization methods are considered for different boosting algorithms. For example, John Duchi and Yoram Singer studied penalties for Adaboost based on the l_1, l_2, and l_∞ norms of the predictor and introduce mixed-norm penalties that build upon the initial penalties[1]. Tong Zhang and Bin Yu exploited early stopping to regularize boosting fitting [10]. Jerome Friedman used "shrinkage" to deal with regularization [2]. By adding a spatial regularization kernel to a standard loss function formulation of the boosting problem, James Xiang Zhang *et al.* added a spatial regularization kernel to a standard loss function formulation to regularize boosting for classification problem [8].

Here, we particularly interest in impose sparsity on one of the most popular boosting algorithm for regression, called gradient boosting algorithm. To avoid overfitting, many regularization methods were considered in other literatures. For example, Hastie *et al.* modified gradient boosting algorithm by impose regularization using shrinkage [4]. Friedman proposed another algorithm "stochastic gradient boosting" to deal with regularization [3]. To be more specific, he proposed that at each iteration of the algorithm, a base learner should be fit on a subsample of the training set drawn at random without replacement, where subsample size is some constant fraction of the size of the training set. Smaller values of this constant fraction introduce randomness into the algorithm and help prevent overfitting, acting as a kind of regularization.

Our main contribution is to propose a new algorithm which directly solves the l_1 regularized loss minimization problem. l_1 regularization has been extensively studied and understood in many literatures, such as [7, 6, 5]. Use l_1 regularization to impose sparsity can lead to a parsimonious model that uses a small number of parameters. Enforcing sparsity may detect the most discriminative information and be a way to avoid overtting. In this paper, We compare this algorithm with gradient boosting theoretically and numerically. We will show that gradient boosting can be viewed as a

degenerate case of our new algorithm. Our experiments on real world data show show that our algorithm achieves better generalization than gradient boosting with sparser composite hypotheses.

2 Preliminaries

2.1 Boosting algorithms for regression

2.1.1 Regression problems

Let $\mathcal{X}$ denote the input space and $\mathcal{Y}$ a measurable subset of $\mathcal{R}$. In regression problems, we assume that the examples are chosen randomly from the same but unknown distribution $\mathcal{D}$. During training, a learning algorithm receives a labeled training set $S = ((x_1, y_1), (x_2, y_2), ..., (x_m, y_m)) \in (\mathcal{X} \times (\mathcal{Y}))^m$ with $y_i = c(x_i)$. The output of the learning algorithm is a prediction rule called a hypothesis $h \in \mathcal{H}$, which can be treated as a function mapping $\mathcal{X}$ to $\mathcal{Y}$.

To measure the quality of a given hypothesis, we use loss function $L : \mathcal{Y} \times \mathcal{Y} \to \mathcal{R}_+$ to measure the magnitude of error(the difference between the real-valued label predicted and the true value).

Given a hypothesis space $\mathcal{H}$, the goal of regression is to find a hypothesis $h \in \mathcal{H}$ with small expected loss with respect to the target f: $E_{\mathcal{D}}[L(c(x), h(x))]$. The empirical loss is defined as $\frac{1}{m}\sum_{i=1}^{m} L(y_i, h(x_i))$.

2.1.2 Gradient Boosting

The goal of boosting for regression is to find a function $F(x) \in \{\sum_{i=1}^{T} \alpha_i h_i(x) + const \big| h \in \mathcal{H}\}$ to approximate the true value $y = c(x)$ so as to minimize the expected value of some specified loss function L(y, F(x)). Here $\mathcal{H}$ is called weak hypothesis space, training and test examples are i.i.d. from the same yet unknown distribution $\mathcal{D}$ and y is real valued.

Gradient Boosting tries to find an approximation $\hat{F}(x)$ that minimizes the average value of the loss function on the training set. It does so by starting with a model, consisting of a constant function $F_0(x)$, and incrementally expanding it in a greedy fashion:

$$F_0(x) = \arg\min_{\alpha} \sum_{i=1}^{m} L(y_i, \alpha)$$

$$F_t(x) = F_{t-1}(x) + \arg\min_{h \in \mathcal{H}} \sum_{i=1}^{m} L(y_i, F_{t-1}(x_i) + h(x_i)).$$

However, there is no simply way to exactly solve the problem of choosing at each step the best h for an arbitrary loss function L, so instead, steepest descent is used. If we only cared about predictions at the points of the training set, and h were unrestricted, we can view $L(y, F)$ not as a functional of F, but as a function of a vector of values $(F(x_1), F(x_2), ..., F(x_m))$. And then we can calculate the gradient of L: $r_{i,t} = -\left[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}\right]_{F(x)=F_{t-1}(x)}$ which is referred to as pseudo-residuals and find step size that minimizes the loss along the negative gradient direction. But as h must come from a restricted class $\mathcal{H}$ of functions we'll just choose the one that most closely approximates the gradient of L (that's what "fit a weaker learner $h_t(x)$ to pseudo-residuals" in gradient boosting pseudocode is doing).

Algorithm 1: Gradient Boosting

input : m examples $(x_1, y_1) \dots (x_m, y_m)$, weak learning algorithm A

$F_0(x) = \text{argmin}_{\alpha} \sum_{i=1}^{m} L(y_i, \alpha)$
for $t = 1$ *to* T **do**
  pseudo-residuals $r_{i,t} = -\left[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}\right]_{F(x)=F_{t-1}(x)}$, $\forall i = 1, \dots, m$
  fit a weaker learner $h_t(x)$ to pseudo-residuals, i.e. train it using the training set$\{(x_i, r_{i,t})\}_{i=1}^{m}$
  $\alpha_t = \text{argmin}_{\alpha} \Sigma_{i=1}^{m} L(y_i, F_{t-1}(x_i) + \alpha h_t(x_i))$
  $F_t(x) = F_{t-1}(x) + \alpha_t h_t(x)$
end
output: $F_T(x)$

Loss Function of Gradient Boosting

$$\mathcal{L}(\boldsymbol{\alpha}) = \sum_{i=1}^{m} L(y_i, \sum_{t=1}^{T} \alpha_t h_t(x_i) + const)$$

2.2 Sparse Representation

Consider a linear prediction model that the target composite predictor is a sparse combination of a set of weak predictor. The sparse representation is interested in identifying those basic weak predictor. We are interested in the case that the number of weak predictor, such as linear functions, step functions, quadratic functions, etc, is larger then the number of training data we have. In this case, an unconstraint error minimization is inadequate because under the scenario of aggregating the weak predictor, the composite predictor is likely to overfit the data. Therefore, the common practice is to impose a sparsity constraint on the weight $\boldsymbol{\alpha}$ to obtained a regularized problem. We define $\|\boldsymbol{\alpha}\|_0 = \{i : \alpha_i \neq 0\}$. The constraint will be $\|\boldsymbol{\alpha}\|_0 \leq k$, where k is a predefined parameter based on our a priori understanding of the problem. A sparse representation of the composite predictor is easier to store, compute and interpret. More over, a simpler predictor is more likely to generalize well to different input data and get a higher prediction accuracy.

3 Adding Sparsity Constraint

The output of boosting algorithm is a linear prediction model. Based on the structure of boosting algorithm, it will keep adding weak predictor into the final predictor to minimize the prediction error. In addition to prediction accuracy, sparsity of the composite predictor is a desirable characteristic. The sparsity over here means that there are only relative small amount of nonzero weight assigned to the composite predictor.

4 Sparse Gradient Boosting Regression

Our goal is to solve the regularized loss minimization problem:

$$\min_{\boldsymbol{\alpha}} \mathcal{L}(\boldsymbol{\alpha}) = \sum_{i=1}^{m} L(y_i, \sum_{j=1}^{|\mathcal{H}|} \alpha_j \frac{h_j(x_i)}{\sum_{i=1}^{m} h_j^2(x_i)})$$

$$s.t. \|\boldsymbol{\alpha}\|_1 = \sum_{j=1}^{|\mathcal{H}|} |\alpha_j| \leq C$$

Here, $\sum_{i=1}^{m} h_j^2(x_i)$ can be treat as $\|h_j\|_2$ on training data and $\frac{h_j(x_i)}{\sum_{i=1}^{m} h_j^2(x_i)}$ is a normalized version of weak hypothesis. l_1 regulization on the coefficient of normalized weak hypotheses is more meaningful than simply on the coefficient of original hypotheses.

The idea is to first run Gradient Boosting on training examples until the coefficients α_j's hit the boundary of the above regularization constant and then use "coordinate descent" procedure to solve the constrained optimization problem. To be more specific, at some round t (on which $F_{t-1}(x_i) = \sum_{j=1}^{|\mathcal{H}|} \alpha_j \frac{h_j(x_i)}{\sum_{i=1}^{m} h_j^2(x_i)}$), consider all possible adjustments along two "coordinates": for any two indices j_1, j_2, we can subtract some $a/2$ from coefficient α_{j_1} and add it to α_{j_2} to keep $\sum_{j=1}^{|\mathcal{H}|} |\alpha_j| \leq C$.

Under different loss functions, we can derive different regularized algorithms. For the rest of our work, we choose squared loss as a demonstration. Our proposed algorithm Sparse Gradient Boosting for $L(y, F) = (y - F)^2/2$ is defined as follows:

Algorithm 2: Sparse Gradient Boosting

input : m examples $(x_1, y_1) \dots (x_m, y_m)$, weak learning algorithm A

$F_0(x) = \text{argmin}_\alpha \sum_{i=1}^m L(y_i, \alpha)$

for $t = 1$ *to* T **do**

$r_{i,t} = -\left[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}\right]_{F(x)=F_{t-1}(x)}$, $\forall i = 1, \dots, m$

$h_{t_{max}} = \text{argmax}_{h_j} \frac{\sum_{i=1}^m h(x_i) r_{i,t}}{\|h\|_2}$

$h_{t_{min}} = \text{argmin}_{h_j : \alpha_j > 0} \frac{\sum_{i=1}^m h(x_i) r_{i,t}}{\|h\|_2}$

$\epsilon = \min(2\alpha_{t_{min}}, \frac{2\sum_{i=1}^m r_i (\frac{h_{t_{max}}(x_i)}{\|h_{t_{max}}\|_2} - \frac{h_{t_{min}}(x_i)}{\|h_{t_{min}}\|_2})}{\sum_{i=1}^m (\frac{h_{t_{max}}(x_i)}{\|h_{t_{max}}\|_2} - \frac{h_{t_{min}}(x_i)}{\|h_{t_{min}}\|_2})^2})$

$\alpha_{t_{max}} \leftarrow \alpha_{t_{max}} + \frac{\epsilon}{2}$

$\alpha_{t_{min}} \leftarrow \alpha_{t_{min}} - \frac{\epsilon}{2}$

$F_t(x) = \sum_{i=1}^t \alpha_i \frac{h_i(x)}{\sum_{i=1}^m h_j^2(x_i)}$

end

output: $F_T(x)$

5 Analysis

5.1 Use standard gradient boosting, then using the solution of the standard gradient boosting as warm-start for the sparse gradient boosting

(1) α will be at the boundary of the constraint OR (2) α is at the interior of the constraint set If (2) then it means that the composite predictor perfectly fits the data, which might leads to over fitting the data, consider reduce C. Therefore, we assume that after running the standard gradient boosting, $\boldsymbol{\alpha}$ will lie on the boundary of the constraint set.

5.2 How do we prove that the algorithm is solving the above mentioned l_1 constraint optimization problem?

We use $\mathcal{L}_{j_1 \to j_2}(\boldsymbol{\alpha}, a)$ to denote the value of loss function after an adjustment on j_1 and j_2, where $\mathcal{L}_{j_1 \to j_2}(\boldsymbol{\alpha}, a) = \mathcal{L}(\alpha_1, \alpha_2, ..., \alpha_{j_1} + \frac{a}{2}, ..., \alpha_{j_2} - \frac{a}{2}, ..., \alpha_{|\mathcal{H}|})$.

The following theorem shows that in the "coordinate descent" procedure, Sparse Gradient Boosting algorithm chooses the adjustment on two coordinates which gives the largest gradient descent and the step size a is chosen to achieve the minimum along those directions.

Theorem 1. *In each iteration, the choice of k, l, and s satisfy the following properties*

$$(l, k) = \textit{argmax}_{(j_1, j_2) : \alpha_{j_1} > 0} \left(- \frac{\partial \mathcal{L}_{j_1 \to j_2}(\boldsymbol{\alpha}, a)}{\partial a} \right)$$

$$\epsilon = \textit{argmin}_{a : a/2 \le \alpha_l} \mathcal{L}_{l \to k}(\boldsymbol{\alpha}, a)$$

Taking l_2 loss as the loss function,

$$\mathcal{L}_{j_1 \to j_2}(\alpha, a) = \frac{1}{2} \sum_{i=1}^m (y_i - \sum_{j=1}^{|\mathcal{H}|} \alpha_j h_j(x_i) + \frac{a}{2} h_{j_1}(x_i) - \frac{a}{2} h_{j_2}(x_i))^2$$

$$- \frac{\partial \mathcal{L}_{j_1 \to j_2}(\alpha, a)}{\partial a} = \sum_{i=1}^m \left((y_i - \sum_{j=1}^{|\mathcal{H}|} \alpha_j \frac{h_j(x_i)}{\|h_j\|_2} + \frac{a}{2} \frac{h_{j_1}(x_i)}{\|h_{j_1}\|_2} - \frac{a}{2} \frac{h_{j_2}(x_i)}{\|h_{j_1}\|_2}) (\frac{1}{2} \frac{h_{j_2}(x_i)}{\|h_{j_1}\|_2} - \frac{1}{2} \frac{h_{j_1}(x_i)}{\|h_{j_2}\|_2}) \right)$$

Letting $a = 0$

$$- \frac{\partial \mathcal{L}_{j_1 \to j_2}(\alpha, a)}{\partial a} \bigg|_{a=0} = \sum_{i=1}^m \left((y_i - \sum_{j=1}^{|\mathcal{H}|} \alpha_j \frac{h_j(x_i)}{\|h_j\|_2}) (\frac{1}{2} \frac{h_{j_2}(x_i)}{\|h_{j_1}\|_2} - \frac{1}{2} \frac{h_{j_1}(x_i)}{\|h_{j_1}\|_2}) \right)$$

$$= \frac{1}{2}\frac{\sum_{i=1}^{m}\left(h_{j_2}(x_i)(y_i - \sum_{j=1}^{|\mathcal{H}|}\alpha_j \frac{h_j(x_i)}{\|h_j\|_2})\right)}{\|h_{j_2}\|_2} - \frac{1}{2}\frac{\sum_{i=1}^{m}\left(h_{j_1}(x_i)(y_i - \sum_{j=1}^{|\mathcal{H}|}\alpha_j \frac{h_j(x_i)}{\|h_j\|_2})\right)}{\|h_{j_1}\|_2}$$

To maximize this quantity under the constraint $\alpha_{j_1} > 0$, we must choose

$$j_2 = \text{argmax}_j \frac{\sum_{i=1}^{m}\left(h_j(x_i)(y_i - \sum_{k=1}^{|\mathcal{H}|}\alpha_k \frac{h_k(x_i)}{\|h_k\|_2})\right)}{\|h_j\|_2},$$

$$j_1 = \text{argmin}_{j,\alpha_j>0} \frac{\sum_{i=1}^{m}\left(h_j(x_i)(y_i - \sum_{k=1}^{|\mathcal{H}|}\alpha_k \frac{h_k(x_i)}{\|h_k\|_2})\right)}{\|h_j\|_2}.$$

The residual r_i under l_2 loss is:

$$r_i = -\left[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}\right]_{F(x)=F_{t-1}(x)} = (y_i - F(x_i)),$$

where $F(x) = \sum_{j=1}^{|\mathcal{H}|}\alpha_j \frac{h_j(x)}{\|h_j\|_2}$ and hence

$$j_2 = \text{argmax}_j \frac{\sum_{i=1}^{m} h_j(x_i) r_i}{\|h_j\|_2}$$

$$j_1 = \text{argmin}_j \frac{\sum_{i=1}^{m} h_j(x_i) r_i}{\|h_j\|_2}$$

Set $\frac{\partial \mathcal{L}_{j_1 \to j_2}(\alpha, a)}{\partial a} = 0$, we get:

$$= \sum_{i=1}^{m}\left(\frac{a}{4}(\frac{h_{j_2}(x_i)}{\|h_{j_2}\|_2} - \frac{h_{j_1}(x_i)}{\|h_{j_1}\|_2})^2 - (y_i - \sum_{j=1}^{|\mathcal{H}|}\alpha_j \frac{h_j(x_i)}{\|h_j\|_2})(\frac{1}{2}\frac{h_{j_2}(x_i)}{\|h_{j_2}\|_2} - \frac{1}{2}\frac{h_{j_1}(x_i)}{\|h_{j_1}\|_2})\right) = 0$$

$$a = \frac{2\sum_{i=1}^{m} r_i(\frac{h_{j_2}(x_i)}{\|h_{j_2}\|_2} - \frac{h_{j_1}(x_i)}{\|h_{j_1}\|_2})}{\sum_{i=1}^{m}(\frac{h_{j_2}(x_i)}{\|h_{j_2}\|_2} - \frac{h_{j_1}(x_i)}{\|h_{j_1}\|_2})^2}$$

5.3 Compare with gradient boosting

Before we start to run Sparse Gradient Boosting algorithm, we first run Gradient Boosting until the coefficients hit the regularization boundary. Assume at this point, $F(x) = \sum_{j=1}^{\mathcal{H}}\alpha' h_j(x) = \sum_{j=1}^{\mathcal{H}}\alpha \frac{h_j(x)}{\|h_j(x)\|_2}$, with $\sum_{j=1}^{\mathcal{H}}\alpha_j = C$. We then compare the performance of continuing Gradient Boosting and switching to Sparse Gradient Boosting after this point. Unlike Gradient Boosting which only adds weak hypotheses, Sparse Gradient Boosting can also reduce the weight of the worst active hypothesis and transfer the weighted from it to the best one, keeping the coefficients within the regularization boundary (which imposes sparsity on the solution).

Sparse Gradient Boosting is indeed a natural extension of Gradient Boosting. Recall that in Gradient Boosting algorithm, the second step in each iteration t is to fit a weaker learner $h_t(x)$ to pseudo-residual $r_{i,t}$. Different weak learning algorithms, e.g. least-square estimation, will choose different h_t. Here, we choose $h_t = \text{argmax}_h \frac{\sum_{i=1}^{m} h(x_i) r_{i,t}}{\|h\|_2}$, which is the same as $h_{t_{max}}$ chosen by Sparse Gradient Boosting. Next, we'll first show why choosing weak hypothesis this way is meaningful, and then we'll show that under this weak learner, each iteration step of Gradient Boosting can be viewed as a degenerate case of Sparse Gradient Boosting.

First, as we mentioned previously, in Gradient Boosting, we want to choose the weak hypothesis h_t that most closely approximates the gradient of L which is r_t at time t. As before, we view $L(y, F)$ not as a functional of F, but as a vector of values $< F(x_1), F(x_2), ..., F(x_m) >$. Let $\boldsymbol{h} =< h(x_1), h(x_2), ..., h(x_m) >$ and residual vector $\boldsymbol{r}_t =< r_{1,t}, r_{2,t}, ...r_{m,t} >$, then we can rewrite the hypothesis we choose on round t as $h_t = \text{argmax}_h \frac{\sum_{i=1}^{m} h(x_i) r_{i,t}}{\|h\|_2} = \text{argmax}_h \frac{\boldsymbol{h} \cdot \boldsymbol{r}_t}{\|\boldsymbol{h}\|_2 \|\boldsymbol{r}_t\|_2}$. Note that $\frac{\boldsymbol{h} \cdot \boldsymbol{r}_t}{\|\boldsymbol{h}\|_2 \|\boldsymbol{r}_t\|_2}$ equals the cosine of the angle between these two vectors. The larger this

quantity, the smaller the angle, and the closer the two vectors. In particular, if it equals to 1, $\boldsymbol{h}$ is parallel to $\boldsymbol{r}_t$, then as in Gradient Boosting, $\boldsymbol{h}$ is a perfect substitute of $\boldsymbol{r}_t$, which is the negative gradient of L. And here, we want all the α_j's are greater than or equal to 0, so when the angle between $\boldsymbol{h}$ and $\boldsymbol{r}_t$ equal 180^o, it is not preferable. In order to achieve this, we design the hypothesis space such that for each hypothesis $h \in \mathcal{H}$, $-h$ is also in $\mathcal{H}$.

Next, we'll show that when we choose h_t like this on each round, the Gradient Boosting can be viewed as degenerate case of Sparse Boosting. To see this first recall that after choosing h_t, we choose $\alpha_t = \text{argmin}_\alpha \Sigma_{i=1}^m L(y_i, F_{t-1}(x_i) + \alpha h_t(x_i))$ and then add $\alpha_t h_t(x)$ to current F_{t-1}. Under l_2 loss, we can derive a closed-form expression of α_t. $\Sigma_{i=1}^m L(y_i, F_{t-1}(x_i) + \alpha h_t(x_i)) = \Sigma_{i=1}^m (y_i - F_{t-1}(x_i) - \alpha h_t(x_i))^2 = \sum_{i=1}^m (r_{i,t} - \alpha h_t(x_i))^2$. We take first derivative with respect to α and set it to 0, we get $\alpha_t = \frac{\sum_{i=1}^m r_{i,t} h(x_i)}{\sum_{i=1}^m h^2(x_i)} = \frac{\sum_{i=1}^m r_{i,t} h(x_i)}{\|h_t\|_2}$. Then Gradient Boosting add $\alpha_t h_t(x) = \sum_{i=1}^m r_{i,t} h_t(x_i) \frac{h_t(x)}{\|h_t\|_2}$ to $F_{t-1}(x)$. In Sparse Gradient Boosting, $h_{t_{max}}$ is the same as h_t in Gradient Boosting on the same round. Now if we remove the requirement that $\alpha_{t_{min}} > 0$, which means that $h_{t_{min}}$ is not necessary an active hypothesis, then the worst hypothesis is just the negation of the best one $h_{t_{max}}$ (from the selection of hypothesis space, we know that the negation of the best h also belongs to $\mathcal{H}$). If we allow the coeffecient of the worst hypothesis to be negative, then we can just choose the step size $\epsilon = \frac{2\sum_{i=1}^m r_{i,t}(\frac{h_{t_{max}}(x_i)}{\|h_{t_{max}}\|_2} + \frac{h_{t_{max}}(x_i)}{\|h_{t_{max}}\|_2})}{\sum_{i=1}^m (\frac{h_{t_{max}}(x_i)}{\|h_{t_{max}}\|_2} + \frac{h_{t_{max}}(x_i)}{\|h_{t_{max}}\|_2})^2} = \frac{\sum_{i=1}^m r_{i,t} h_{t_{max}}(x_i)}{\sum_{i=1}^m h_{t_{max}}^2(x_i)} \|h_{t_{max}}\|_2 = \sum_{i=1}^m r_{i,t} h_{t_{max}}(x_i)$. Then in Sparse Gradient Boosting algorithm, we add $\frac{\epsilon}{2} \frac{h_{t_{max}}(x)}{\|h_{t_{max}}\|_2} = \frac{1}{2} \sum_{i=1}^m r_{i,t} h_{t_{max}}(x_i) \frac{h_{t_{max}}(x)}{\|h_{t_{max}}\|_2}$ and substract $\frac{\epsilon}{2} \frac{h_{t_{min}}(x)}{\|h_{t_{min}}\|_2} = -\frac{\epsilon}{2} \frac{h_{t_{max}}(x)}{\|h_{t_{max}}\|_2}$ to $F_{t-1}(x)$. Since $h_t = h_{t_{max}}$, those two algorithms are adding the same amount of the same weak hypothesis to $F_{t-1}(x)$. That's to say, Gradient Boosting can be viewed as a degenerate case of Sparse Gradient Boosting algorithm.

In order to clearly illustrate the above mentioned comparison, we rewrite gradient boosting in a similar way as sparse gradient boosting algorithm and put them together as follows:

Gradient Boosting

input : m examples $(x_1, y_1) \dots (x_m, y_m)$,
weak learning algorithm A
$F_0(x) = \text{argmin}_\alpha \sum_{i=1}^m L(y_i, \alpha)$

for $t = 1$ *to* T **do**
- $r_{i,t} = -\left[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}\right]_{F(x)=F_{t-1}(x)}$
- $h_t = \text{argmax}_{h_j} \frac{\sum_{i=1}^m h(x_i) r_{i,t}}{\|h\|_2}$
- $\epsilon = \frac{2\sum_{i=1}^m r_{i,t} h(x_i)}{\sum_{i=1}^m h^2(x_i)}$
- $\alpha_t \leftarrow \alpha_t + \frac{\epsilon}{2}$
- $F_t(x) = \sum_{i=1}^t \alpha_i \frac{h_i(x)}{\sum_{i=1}^m h_j^2(x_i)}$

end
output: $F_T(x)$

Sparse Gradient Boosting

input : m examples $(x_1, y_1) \dots (x_m, y_m)$,
weak learning algorithm A
$F_0(x) = \text{argmin}_\alpha \sum_{i=1}^m L(y_i, \alpha)$

for $t = 1$ *to* T **do**
- $r_{i,t} = -\left[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}\right]_{F(x)=F_{t-1}(x)}$
- $h_{t_{max}} = \text{argmax}_{h_j} \frac{\sum_{i=1}^m h(x_i) r_{i,t}}{\|h\|_2}$
- $h_{t_{min}} = \text{argmin}_{h_j:\alpha_j>0} \frac{\sum_{i=1}^m h(x_i) r_{i,t}}{\|h\|_2}$
- $\epsilon =$ $\min(2\alpha_{t_{min}}, \frac{2\sum_{i=1}^m r_i(\frac{h_{t_{max}}(x_i)}{\|h_{t_{max}}\|_2} - \frac{h_{t_{min}}(x_i)}{\|h_{t_{min}}\|_2})}{\sum_{i=1}^m (\frac{h_{t_{max}}(x_i)}{\|h_{t_{max}}\|_2} - \frac{h_{t_{min}}(x_i)}{\|h_{t_{min}}\|_2})^2})$
- $\alpha_{t_{max}} \leftarrow \alpha_{t_{max}} + \frac{\epsilon}{2}$
- $\alpha_{t_{min}} \leftarrow \alpha_{t_{min}} - \frac{\epsilon}{2}$
- $F_t(x) = \sum_{i=1}^t \alpha_i \frac{h_i(x)}{\sum_{i=1}^m h_j^2(x_i)}$

end
output: $F_T(x)$

6 Experiment

1. iterations vs number of weak predictor
2. iterations vs generalization/training error

3. sparsity parameter k vs generalization error under T iterations + constant line of generation error under T iterations w/o sparsity constraint

7 Conclusion

References

References follow the acknowledgments. Use unnumbered third level heading for the references. Any choice of citation style is acceptable as long as you are consistent. It is permissible to reduce the font size to 'small' (9-point) when listing the references. **Remember that this year you can use a ninth page as long as it contains *only* cited references.**

References

[1] John Duchi and Yoram Singer. Boosting with structural sparsity. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 297–304. ACM, 2009.

[2] Jerome H. Friedman. Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29:1189–1232, 2000.

[3] Jerome H Friedman. Stochastic gradient boosting. *Computational Statistics & Data Analysis*, 38(4):367–378, 2002.

[4] T Hastie, R Tibshirani, and J. H Friedman. 10. boosting and additive trees. In *The Elements of Statistical Learning*, pages 337–384. New York: Springer, second edition, 2009.

[5] Andrew Y Ng. Feature selection, l 1 vs. l 2 regularization, and rotational invariance. In *Proceedings of the twenty-first international conference on Machine learning*, page 78. ACM, 2004.

[6] Mark Schmidt. Least squares optimization with l1-norm regularization. *Project Report, University of British Columbia*, 2005.

[7] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[8] Z Xiang, Y Xi, Uri Hasson, and P Ramadge. Boosting with spatial regularization. *Advances in Neural Information Processing Systems*, 22:2107–2115, 2009.

[9] Tong Zhang. Adaptive forward-backward greedy algorithm for sparse learning with linear models. NIPS, 2008.

[10] Tong Zhang and Bin Yu. Boosting with early stopping: Convergence and consistency. *Annals of Statistics*, pages 1538–1579, 2005.